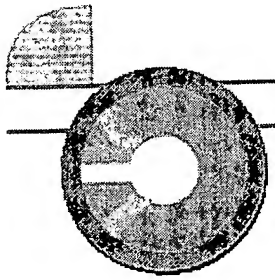


port knocking

Helping you keep sensitive data accessible and protected.

A STEALTHY SYSTEM FOR NETWORK AUTHENTICATION ACROSS CLOSED PORTS

about

firewall primer

details

knock lab

download

documentation

FAQ

logos & banners

contribute code

resources

contact

Articles about port knocking have appeared in the Linux Journal (16 June 2003) and SysAdmin Magazine (June 2003).

LINUX JOURNAL

DETAILS OF PORT KNOCKING MECHANISM

Once you've perused the **firewall primer**, learn about the detai of port knocking here. Ideas about how to use port knocking in simple situations are presented, as well as an outline of how to use encryption to avoid eavesdropping.

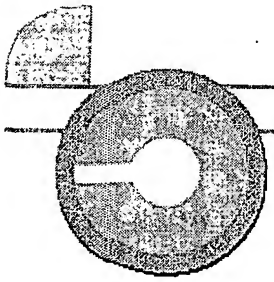
overview | application | spoofing | transfer | download

DATA TRANSFER ACROSS CLOSED PORTS

Information is flowing across closed ports to modify firewall process can be extended to transfer any kind of information closed ports and generalizes to the idea of data transfer acr ports. The data to be transfered can be embedded in a knoc following way.

```
{header} {payload info} {payload} {checksum}
```

where each `{section}` is comprised of one or more individu For example, the `{header}` and `{footer}` can be designed t distinguish a data knock from an authentication knock. The `info}` sequence could store the number of entries in `{payl` some other useful information. The `{payload}` would be yo stream encoded, and possibly encrypted, into an integer se Finally, `{checksum}` could be used to contain information w to validate the integrity of the received payload.



port knocking

Helping you keep sensitive data accessible and protected.

DETAILS OF PORT KNOCKING MECHANISM

Once you've perused the **firewall primer**, learn about the detai of port knocking here. Ideas about how to use port knocking in simple situations are presented, as well as an outline of how to use encryption to avoid eavesdropping.

overview | application | spoofing | transfer | download

DOWNLOAD PERL PROTOTYPE

The **Perl prototype** implements knock encryption using the Blowfish algorithm in the manner described in this example. There is no initialization vector to keep the knock sequence short (8 unsigned chars). Adding an initalization vector increases the length of the sequence but ensures that repeated encryption of the same information will yield different sequences.

about

firewall primer

details

knock lab

download

documentation

FAQ

logos & banners

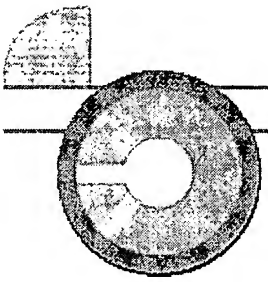
contribute code

resources

contact

Articles about port knocking have appeared in the Linux Journal (16 June 2003) and SysAdmin Magazine (June 2003).

LINUX JOURNAL



port knocking

Helping you keep sensitive data accessible and protected.

A STEALTHY SYSTEM FOR NETWORK AUTHENTICATION ACROSS CLOSED PORTS

about

firewall primer

details

knock lab

download

documentation

FAQ

logos & banners

contribute code

resources

contact

Articles about port knocking have appeared in the Linux Journal (16 June 2003) and SysAdmin Magazine (June 2003).

LINUX JOURNAL

PORT KNOCKING

Here you can **learn** about firewalls and **discover** port knocking, find out how to use port knocking to **secure** your servers with a experimental **Perl prototype**, **construct** knock sequences, or **contribute** to the port knocking project, and see what others ar saying about Port Knocking on **comp.os.linux.advocacy**. A number of individuals suggested that port knocking is a form of **security through obscurity** - check the **author's reply**.

summary | features | port forwarding | port triggering | obscurity | requir
post scriptum | jokes

SECURITY THROUGH OBSCUIRITY AND PORT KNOCKING

It has been **pointed out by some** that port knocking is **a** form of **security through obscurity**.

The concept of security through obscurity is **well describec** by Jay Beale of the Bastille Linux Project. This article was written independently of port knocking and I'm using it here to help define the notion of obscurity. Jay writes (emphasis my own)

> *First, what does the security professional mean by bac "security through obscurity?" We really mean "securit implemented solely through obscurity." This describes the state where your entire method of security resides in hoping that the attacker doesn't know something about the setup of your network, computer or progran One simple case is where you put your company's secrets on an internal webserver, with no password-protection on the pages. Instead of relying on passwo of access control, you're relying on something differen know about that webserver except for the internal cor This almost seems like a decent assumption, except tl discovery tools (like cheops, firewalk, snmpwalk and r webserver on your network.* ***See, the problem is tha the data's location as your sole method of acces***